Image Defogging Using Conditional Denoising Diffusion Probabilistic Models (DDPMs)

Nidhin Harilal*
University of Colorado Boulder
Boulder, CO, USA

Nitin Kumar
University of Colorado Boulder
Boulder, CO, USA

Tushar Gautam
University of Colorado Boulder
Boulder, CO, USA

Abstract

Fog and haze have long been detrimental factors in obtaining clear and high-quality images, significantly affecting computer vision and image processing applications. Traditional single-image defogging methods often rely on the physical scattering model or utilize machine learning-based techniques. However, these methods have been plagued with limitations, such as the inability to recover fine details, susceptibility to noise, and poor generalization. In this paper, we propose a novel single-image defogging method based on the denoising diffusion probabilistic model, which addresses the shortcomings of existing techniques. Our experiments on Cityscapes benchmark dataset show promising results.

1 Introduction

Bad weather, sometimes brings with itself fog, which causes corruption/degradation in image quality leading to loss of visual information from the image for ex. colour, contrast, details of artifacts etc. It directly affects the performance of Outdoor video monitoring system, Visual systems of self driving cars, Remote sensors, etc. It also causes increasing road accidents.

Image defogging is a technique used in computer vision to remove fog or haze from an image to reintroduce the visual clarity and the artifacts lost due to fog. Existing methods either rely on physical model-based restoration [18, 10], which comes with assumptions underlying the degradation process or utilize image processing-based enhancement techniques [19, 4], only tweaking elements like brightness and contrast of the image to improve their visual effect. [18] presents an algorithm which is based on physical model and maximum entropy theory for the deweathering of the image. It introduces a dichromatic atmospheric scattering model to improve the image. Atmospheric degradation based models extract the priori information of the foggy images and use Dark channel prior method and estimate fog density [6]. As discussed above, Physical degradation models make use of a priori knowledge to compute scene depth of the image. It also affects the performance of defogging since a priori knowledge of physical degradation model is not generalizable to any scene and thus in effect the scene depth information becomes unstable. Image processing based techniques include homomorphic filtering for color image enhancement [17], histogram equalization [12], wavelet transform [5] etc. These methods are able to remove fog and haze but also change the image visuals. They are ineffective they create false foreground which are not useful for the practical purposes.

Owing to the complexity of reconstructing images from foggy signals, we leverage denoising diffusion models [7]- a recent emerging topic in computer vision, demonstrating remarkable results in generative modeling [3]. In this project, we implement a conditional denoising diffusion probabilistic model as a dehazing method for image defogging. Denoising diffusion models are a class of probabilistic generative models defined by a Markov chain of diffusion steps to slowly add random noise to data and then learn to reverse the diffusion process to construct desired data samples from the noise [15]. Diffusion models are capable of generating images either conditionally [11] or unconditionally [8]. Unconditional image generation simply means that the model converts noise into any random representative data sample. In this project, we exploit

*Corresponding author: Nidhin Harilal, nidhin.harilal@colorado.edu

conditional denoising diffusion models that make the denoising process conditional on an input foggy signal. Our conditional denoising diffusion model is a parameterized Markov chain trained using variational inference to dehaze foggy samples from the data. Unlike the previous methods, They are able to construct the image foreground effectively keeping up the visual artifacts intact along with the removal of the fog. They are highly generalizable to a large set of natural images covering different weather conditions.

2 Dataset

The Dataset used are Cityscapes [1] and Foggy Cityscapes [14] which contains 19998, 8-bit images of urban street scenes and its corresponding synthetic foggy images respectively. Cityscapes dataset focuses on semantic understanding of urban street scenes. It contains 30 classes of objects covering 50 cities and multiple weathers (spring, summer, fall). Foggy Cityscapes is derived from Cityscapes dataset using fog simulation.

Figure 1: Foggy Cityscape Example

Figure 2: Clear Cityscape Example

3 Denoising Diffusion Models (DDMs)

Diffusion models are type of generative models which are inspired by the idea of Non-Equilibrium Statistical Physics [9]. It follows this idea and uses Markov chain to gradually transform one distribution to another [15]. It consists of two opposite processes ie. forward and reverse diffusion process. The model is represented by the latent variables of the form $p_\theta(x_0) = \int p_\theta(x_{0:T})\, dx_{1:T}$ where $x_1, x_2, ..., x_T$ are the latent variables which have the same dimensions as input data (image) $x_0 \sim q(x_0)$.

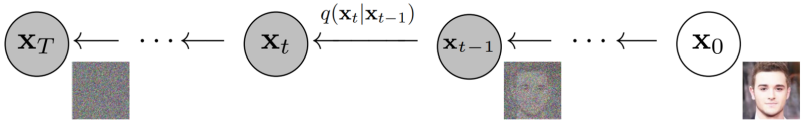

Figure 3: Forward Diffusion Process

During the forward diffusion process shown in Figure 3, noise is added slowly and iteratively which corrupts the image moving away from their existing subspace thus converting a complex image distribution to a simple and known distribution (ex. Gaussian). It is easy to sample from this known distribution. After the completion of the forward process, a simple distribution is obtained mapping each of the training image to a simple outside data space. The forward process has the approximate posterior denoted by $q(x_{1:T}|x_0)$. It follows the Markov chain where noise is added at each step which creates the next state. Noise is added according to a variance schedule denoted by $\beta_1, \beta_2, \beta_3, ..., \beta_T$ such that:

$$q(x_{1:T} \mid x_0) := \prod_{t=1}^{T} q(x_t \mid x_{t-1}) \tag{1}$$

$$q(x_t \mid x_{t-1}) := \mathcal{N}\left(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t I\right) \tag{2}$$

Here, the term β is known as "diffusion rate" and is computed using the variance scheduler. I is the identity matrix and thus distribution at each timestep is Isotropic gaussian. Also, the computation of latent sample

at timesetp t requires t-1 timesteps using the Markov chain following the intermediate states. It is solved by reformulating the kernel to directly go from timestep 0 (ie. the original image) to timestep t as follows:

$$\alpha_t = 1 - \beta_t \tag{3}$$

$$\overline{\alpha}_t = \prod_{s=1}^{t} \alpha_s \tag{4}$$

Using equations above, forward diffusion process can be simplified as follows:

$$q(x_t|x_0) = N(x_t|\sqrt{\overline{\alpha_t}}x_0, (1 - \overline{\alpha_t})I) \tag{5}$$

such that:

$$x_t = \sqrt{\overline{\alpha_t}}x_0 + \sqrt{1 - \overline{\alpha}_t}\epsilon, \quad \epsilon\ N(0, 1) \tag{6}$$

where ϵ is the "noise" that is added iteratively at each step and is sampled from the standard gaussian distribution.

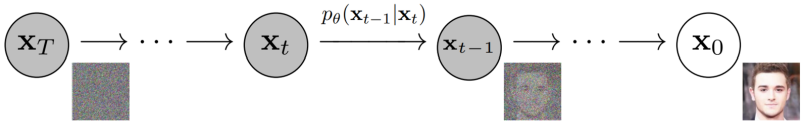

Figure 4: Reverse Diffusion Process

During the reverse diffusion process shown in Figure 4, reversal of the forward diffusion process takes place ie. the corruption created by adding the noise during the forward process is reversed. It starts from a simple distribution outside the data space and it aims to reach the original data space. There can be an infinite number of paths to reach the original data space. It is tackled by referring to the iterative steps taken during the forward process. Thus, Deep learning model is used to predict the pdf parameters of the forward process and is trained to iteratively reach the original data space. The Joint distribution is represented by $p_\theta\ (x_{0:T})$. It follows the Markov chain with learnt Gaussian transition which starts at $p(x_T) = N(x_T; 0, I)$ such that:

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t) \tag{7}$$

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta(x_t, t)) \tag{8}$$

4 Proposed Method

4.1 Conditional Denoising Diffusion Model

Conditional diffusion models are a class of generative models particularly in the context of images. They are built on the foundation of denoising score, which offer a powerful framework for learning distributions over data. In this case, we are interested in learning the transformation between foggy and non-foggy images.

Let X be the space of foggy images and Y be the space of non-foggy images. We are given a dataset $D = \{x_i, y_i\}_{i=1}^{N}$ of N input-output pairs, where $x_i \in X$ and $y_i \in Y$. Our goal is to learn a parametric approximation to the conditional distribution $P(Y|X)$ that generates a clear image y given a foggy input image x. The conditional diffusion model works in two main steps: the forward diffusion process and the reverse diffusion process. The conditional diffusion process trough a stochastic iterative refinement process that maps a foggy image x to a target image $y \in R^d$. We approach this problem by adapting the denoising diffusion probabilistic (DDPM) model of Ho et al.[7] to conditional image generation. Similar to DDPM, the conditional diffusion model works in two main steps: the forward diffusion process and the reverse diffusion process in terms of the involved *random variables*:

- **Forward diffusion process:** Following the mechanism defined in Ho et al.[7], we define the forward Markovian diffusion process q that gradually adds Gaussian noise to the foggy image x_0 over T iterations. Importantly, using some algebraic manipulation, one can rewrite the distribution x_t given x_0 by marginalizing out the intermediate steps as:

$$q(x_t|x_0) = N(x_t|\sqrt{\overline{\alpha_t}}x_0, (1-\overline{\alpha_t})I) \tag{9}$$

  where $\alpha_t = 1 - \beta_t$ (β_t is the noise scheduler) and $\overline{\alpha_t} = \prod_{i=1}^{T} \alpha_i$
- **Reverse diffusion process:** The reverse process aims to denoise the foggy images X back into the non-foggy images Y. For our case, the model gets information about both the noisy foggy image x_t at arbitrary time t and the original non-foggy image x_0. We are interested in deriving a posterior to the parametric approximation of y_{t-1} (noisy non-foggy) given x_t (noisy foggy) and x_0 (original foggy), that is - $p(y_{t-1}|x_t, x_0)$

  This posterior distribution is helpful when parameterizing the reverse chain and formulating a variational lower bound on the log-likelihood of the reverse chain.

Optimizing Conditional Diffusion: We use the original foggy image x_0 as additional side information to aid in reversing the diffusion process, and we optimize a neural denoising model p_θ that uses this non-foggy image x_0 and a noisy target image $\tilde{y}$ (described below) as inputs.

$$\tilde{y} = \sqrt{\overline{\alpha_t}}x_0 + \sqrt{1-\alpha_t}\epsilon, \quad \epsilon\ N(0,1) \tag{10}$$

Our goal here would be to recover the noiseless target image y_0, such that it is compatible with the marginal distribution of noisy images at different steps of the forward diffusion process (defined previously).

4.2 Model Architecture

We define our denoising model $p_\theta(x_0, \tilde{y})$ to have U-Net [13] like architecture following the work [7] with some modifications including re-scaling skip connections by $\frac{1}{\sqrt{2}}$. An example visualization of U-Net architecture is shown in Figure 5. Note that we also add information on time to the U-Net, to let the model know that it's in the t^{th} state of reverse diffusion. U-net can be broadly thought of as an encoder network followed by a decoder network, a structure widely used in vision and applications involving image-to-image tasks.

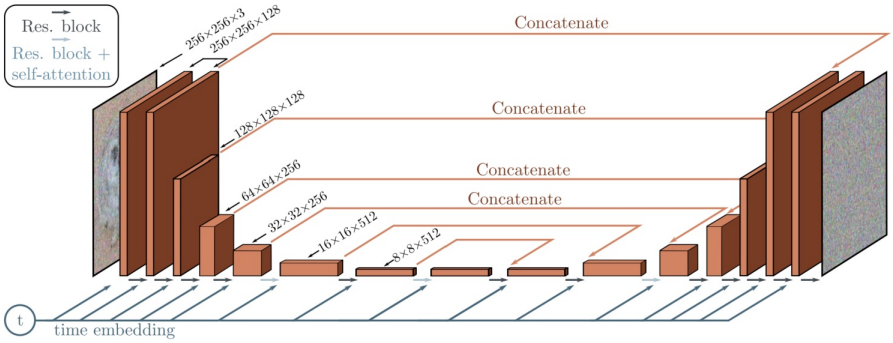

Figure 5: Modified U-Net architecture derived from the original U-Net[13]

Noise Schedule (β_t): For our training, we use a linear noise scheduler that linearly increases from $0.0001 \rightarrow 0.02$ across T timesteps ($T = 1000$).

4.3 Inference

Inference under our model is defined from the reverse Markovian process, starting from Gaussian noise x_T as follows:

$$p_\theta(y_{0:T}|x) = p(x_T)\prod_{t=1}^{T} p_\theta(y_{t-1}|x_t, x_0) \tag{11}$$

$$p(x_T) = N(x_t|0, I) \tag{12}$$

$$p_\theta(y_{t-1}|x_t, x_0) = N(y_{t-1}|\mu_\theta(x, y_T, \beta_T), \sigma_T^2(\beta_T)) \tag{13}$$

The inference process can be defined in terms of isotropic gaussian conditional distributions $p_\theta(y_{t-1}|x_t, x_0)$ which are learned. The idea is that if the noise variance of the forward process steps is very small, the reverse process $p_\theta(y_{t-1}|x_t, x_0)$ would be approximately gaussian. Note that our infernce is *deterministic* as there is no such stochasticity to the parameters of neural network itself. That is, our inference gives us point estimates rather than distributions, therefore, follows a deterministic inference.

5 Experiments

In this section, we present the experimental setup and results of applying our conditional diffusion model to defog images.

5.1 Experimental Settings

To implement our diffusion model, we employed a U-Net architecture as the backbone, as mentioned before. U-Net is a popular choice for image-to-image translation tasks due to its symmetric structure and ability to capture both local and global context [13]. As discussed in the previous section, we opted for a linear noise scheduler for controlling the noise schedule β_t, which ensured a smooth transition from foggy to non-foggy images over 1000 diffusion steps. This enabled the model to learn intricate details and progressively refine the image quality throughout the reverse diffusion process.

Our model was trained for $200k$ iterations using the Adam optimizer, a widely used optimization algorithm known for its adaptability and convergence properties. To stabilize the training process, we incorporated a warm-up phase that gradually increased the learning rate before reaching the target value. As for the loss function, we employed mean squared error (MSE) to measure the discrepancy between the predicted and ground-truth images, thus encouraging the model to produce visually accurate defogged images. We trained our model on an *NVIDIA RTX 3090 GPU*, which provided sufficient computational power to handle the intensive training required for the 1000 diffusion steps.

Implementation Details: The implementation of our conditional diffusion model was carried out using the *PyTorch* and *PyTorch Lightning* frameworks. *PyTorch* enabled us to develop the model with ease and flexibility. At the same time, *PyTorch Lightning* provided a high-level interface for managing the training loop and other processes, simplifying the experimental setup. You can find our project repository on Github[2]

5.2 Results

We observed that our model was successful in producing high-quality defogged images, effectively removing fog from the input images while preserving fine details and structures. Furthermore, the U-Net architecture and linear noise scheduler facilitated a smooth transformation between the foggy and non-foggy images throughout the reverse diffusion process. The combination of a U-Net architecture, linear noise scheduler, and 1000 diffusion steps enabled our model to learn intricate details and generate visually accurate defogged images.

5.3 Existence of visual artifacts

While our conditional diffusion model has demonstrated success in defogging images, we observed occasional visual artifacts in the sky regions of some generated images. These artifacts were not consistently present but appeared sporadically upon rerunning the inference multiple times as shown in Figure 7. There could be several possible explanations for the occurrence of these artifacts. Our hypothesis based on the observed samples was that the diffusion process is inherently sensitive to high-frequency noise. During the reverse diffusion process, high-frequency noise could be amplified, resulting in artifacts that become more noticeable in smooth areas like the sky.

Possible solution to avoiding artifacts: One potential approach to mitigate the occurrence of visual artifacts in sky regions is to employ non-linear noise schedulers. Non-linear noise schedulers, as shown in Figure 8 can adapt the noise schedule β_t in a more sophisticated manner, allowing the model to focus on different aspects of the image at different stages of the reverse diffusion process. Recent literature [2, 16] suggests that non-linear noise schedulers can better handle high-frequency noise, reducing the likelihood of artifacts in smooth areas like the sky.

[2]Project repository: `https://github.com/cryptonymous9/DiffusionDefogging`

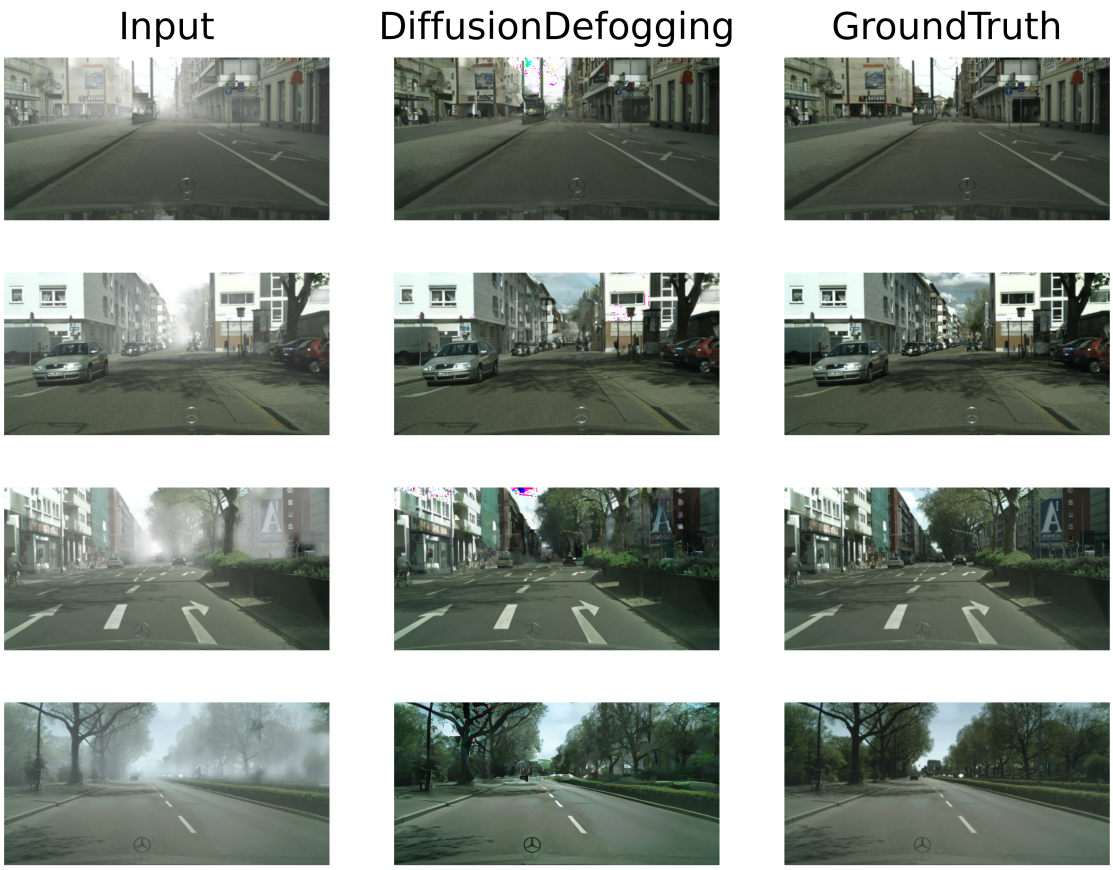

Figure 6: Predictions of our Conditional Diffusion model on the test-set of foggy *Cityscapes*

Figure 7: Results showing the existence of visual artifacts (highlighted in red boxes). DD #i refers to the i^{th} re-run of the model with the same input.

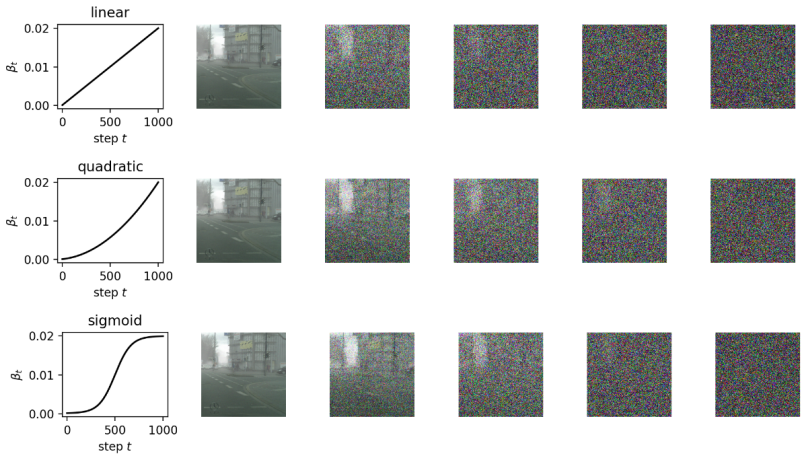

Figure 8: Comparison of the effect of non-linear noise schedulers in forward diffusion.

6 Discussion

While both Markov networks and Belief networks can be used as generative models to learn the probability distribution of a dataset, however in the case of DDMs, Belief networks are more appropriate than Markov networks for several reasons. First, Belief networks can model conditional dependencies between variables in a more expressive way than Markov networks. DDMs require a high degree of flexibility in their model architecture to capture complex patterns in the data. Belief networks can be also easily extended to include additional nodes and edges, which can be used to capture more complex relationships between variables. In contrast, Markov networks have a fixed graph structure, which can limit their expressiveness and make it difficult to capture complex interactions between variables.

Moreover, the inference method used in DDMs does depend on some assumptions. First, the noise is added to the data during the diffusion process. This noise is assumed to be additive white Gaussian noise, with a variance parametrized by the β_t. Second, the conditional probability distribution is assumed to be smooth and continuous. The scheduler can also be treated as a parameter instead of a hyperparameter which then finds an optimal setting for the scheduler for inference.

7 Conclusion and Future Scope

In this project, we have presented a conditional diffusion model for defogging images, leveraging the U-Net architecture and a linear noise scheduler to learn the transformation between foggy and non-foggy images. Our experiments demonstrated the model's ability to generate high-quality defogged images while maintaining visual fidelity in most scenarios. Despite the occasional visual artifacts observed in sky regions, our model serves as a strong baseline for image-defogging tasks. The limitations identified can serve as the foundation for future improvements, such as incorporating non-linear noise schedulers or exploring alternative architectures to handle noise better and recover fine details. Overall, the conditional diffusion model offers a promising approach to image defogging, with the potential for further refinement and optimization. This work has contributed to our understanding of diffusion models and their applications in image-to-image translation tasks, paving the way for future research and advancements in the field.

References

[1] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[3] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[4] Tanghuai Fan, Changli Li, Xiao Ma, Zhe Chen, Xuan Zhang, and Lin Chen. An improved single image defogging method based on retinex. In *2017 2nd international conference on image, vision and computing (ICIVC)*, pages 410–413. IEEE, 2017.

[5] Marie Farge. Wavelet transforms and their applications to turbulence. *Annual Review of Fluid Mechanics*, 24(1):395–458, 1992.

[6] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(12):2341–2353, December 2011.

[7] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[8] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[9] C. Jarzynski. Nonequilibrium equality for free energy differences. *Phys. Rev. Lett.*, 78:2690–2693, Apr 1997.

[10] GUO JIA, WANG XIAOTONG, HU CHENGPENG, and YANG CHANGQING. Simple defogging method for outdoor images based on physical model. In *Proceedings of SPIE, the International Society for Optical Engineering*, volume 7658, 2010.

[11] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.

[12] Stephen M. Pizer, E. Philip Amburn, John D. Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B. Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3):355–368, 1987.

[13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[14] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126(9):973–992, Sep 2018.

[15] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[16] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[17] Liviu I. Voicu, Harley R. Myler, and Arthur Robert Weeks. Practical considerations on color image enhancement using homomorphic filtering. *Journal of Electronic Imaging*, 6(1):108 – 113, 1997.

[18] Xin Wang and Zhenmin Tang. Automatic image de-weathering using physical model and maximum entropy. In *2008 IEEE Conference on Cybernetics and Intelligent Systems*, pages 996–1001. IEEE, 2008.

[19] Yong Xu, Jie Wen, Lunke Fei, and Zheng Zhang. Review of video and image defogging algorithms and related studies on image restoration and enhancement. *Ieee Access*, 4:165–188, 2015.